International Journal of Engineering Trends and Technology
ISSN: 2231-5381/ https://doi.org/10.14445/22315381/IJETT-V74I4P122

Volume 74 Issue 4, 290-298, April 2026

Original Article

Implementation of Fisher-Yates Shuffle Algorithm in Generating Window Card for Elementary School Learners

Eugene S. Valeriano

Tarlac Agricultural University, Camiling, Tarlac, Philippines.

Corresponding Author : esvaleriano@tau.edu.ph

Received: 28 May 2025 Revised: 10 December 2025 Accepted: 28 February 2026 Published: 29 April 2026

Abstract *- The popular method for generating random permutations is the Fisher-Yates Shuffle Algorithm, which guarantees an efficient and effective shuffling method. This study explores its implementation in generating Window Cards for elementary school learners, a learning tool designed to enhance engagement and knowledge retention. By integrating the algorithm, the study aims to create dynamically randomized yet fair distributions of learning content on these cards. The implementation was tested for its effectiveness in producing diverse and non-repetitive sets, ensuring equal exposure to different operations. The Fisher-Yates algorithm significantly improves the process of randomization compared to traditional methods, reducing predictability and enhancing educational effectiveness. It also concludes that applying computational techniques can optimize educational tools, making learning more interactive and unbiased.*

Keywords *- Engagement, Generation, Learning, Randomization, Window card.*

1. Introduction

Randomization is a fundamental aspect of educational tools, especially those designed to enhance engagement and learning outcomes among learners. The Fisher-Yates Shuffle Algorithm is renowned for its efficiency in generating random permutations of a finite array, making it a valuable asset in educational applications for generating questions randomly [1]. Fisher-Yates Shuffle Algorithm adaptability has been shown by recent uses in educational settings. It is used to create instructional games that need to be randomly generated in order to improve learning outcomes. Keeping users from becoming bored, one of the strategies is to employ the randomization technique to give them different patterns with different questions. [2]. On the other hand, the primary objective of any educational institution is to provide high-quality tests that are easy to administer in order to assess learners' comprehension levels. One of the Fisher-Yates Shuffle algorithm techniques can assist with question jumbles. By providing distinct question sets from those of other learners, these algorithms are utilized in numerous research studies to improve educational resources and reinvent conventional methods to lower the chance of cheating [3].

There are several researchers who have developed applications by using randomization techniques [4]. One of which was research conducted by Abdi Suhazli, who used the Fisher-Yates shuffle method to randomly arrange the parts of a picture in a puzzle game. Meanwhile, Widi Aulia Rohmah [5] also conducted research in a quiz game that randomly selects questions; in this way, the user is unable to predict the next question. Alternatively, research conducted by Shafali Agarwal, where random images were used to create the pixel from a permutation key using the Knuth shuffle method and dynamic diffusion.[6]. This study proposes using the Fisher-Yates Shuffle Algorithm to create Window Cards, an interactive learning tool designed to help elementary school learners become more proficient in mathematics. Window Cards are educational tools that display mathematical concepts that the learners can actively engage with the content [7]. Over time, the predictability of traditional static versions of these cards may diminish their efficacy. Meanwhile, by employing the Fisher-Yates Shuffle algorithm in the generation of a window card, every encounter of the learners with the random sets can provide a different arrangement, which keeps learners interested and fosters deeper comprehension. Thus, the significance of randomization in educational settings is not just keeping students interested; it also plays a crucial role in fair assessment and collaborative learning. Social barriers can be removed, and a more inclusive classroom climate can be promoted by randomly allocating tasks to individuals and groups. [8]. Finally, these implementations demonstrate how the algorithm can be used to provide dynamic and captivating educational resources. In order to improve the educational experiences of elementary school learners, this study intends to use the Fisher-Yates Shuffle Algorithm to convert conventional teaching materials into interactive and adaptive learning resources.

2. Objectives of the Study

The objective of the study is to implement an application that integrates the Fisher-Yates Shuffle Algorithm into the generation of Window Cards for mathematical exercises.

Specifically, it aims to;

- Create two random numbers using the algorithm for pairing.
- Assess the algorithms in producing two unique random permutations.
- To implement the algorithm and generate window cards for the following basic math operations.
  - Addition
  - Multiplication
  - Subtraction
  - Division

3. Conceptual Framework

Figure 1 shows the conceptual framework of the implementation of the Fisher-Yates Algorithm in generating a Window Card for elementary School learners. First, the teacher identifies the total items and selects one of the four fundamental operations, like Multiplication, Division, Addition, and Subtraction.

Each selected configurations supply to the system and create randomized numbers with the assessment of unique random permutation incorporated by the Algorithm, which store the data in the database for opted printing of Window Cards and generating mathematical exercise to the learners via web and mobile application to investigate the influence of randomized window card on learners' engagement, collaboration, and assess the numerical level of learners.

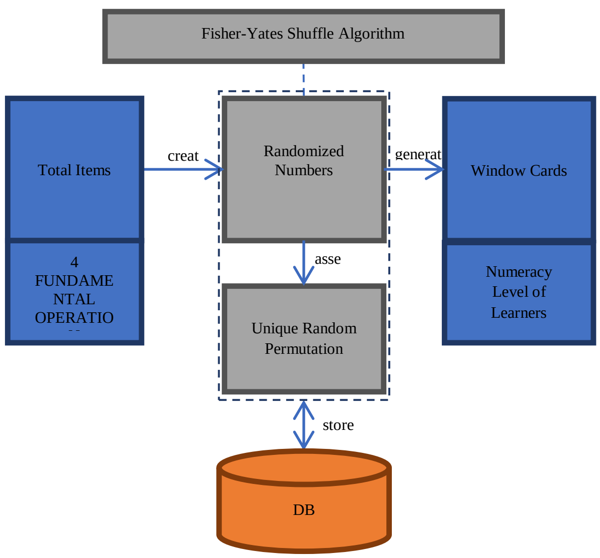

Fig. 1 Conceptual framework of the implementation of the Fisher-Yates Algorithm in generating the Window Card for Elementary School learners

4. Materials and Methods

4.1. Create a Randomized Number using Fisher-Yates Shuffle Algorithm

Figure 2 illustrates the algorithm's methods for generating randomized numbers through a flowchart.

The flowchart illustrates how teachers input data and select which fundamental mathematical operations the program should produce.

After the operations are chosen, it will use the Fisher-Yates technique to generate an unbiased permutation of pair numbers, randomly selecting one or two digit sets every number of items generated in each quarter.

The paired numbers that were generated resulted in a set of Window Cards that the teacher may print off and use to evaluate the learners.

Moreover, the method has the ability to get the entry of a teacher on how many items to be generated by the application. Initially, the common number of items generated is 100 items by default.

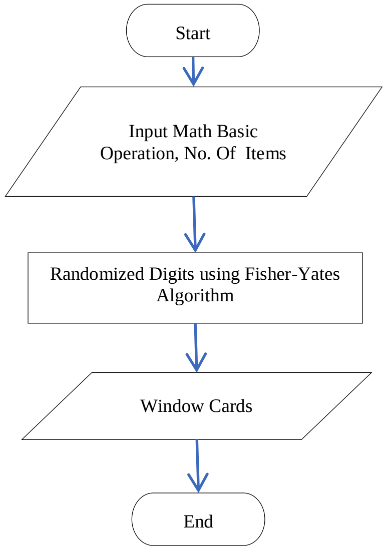

Fig. 2: Flow Chart for generating a randomized number using the Fisher-Yates Algorithm

4.2. Assessed Unique Random Permutation

In Figure 3, the diagram evaluates distinct random permutations generated by the method and, at the end, determines whether each pair contains a duplicate pair in the other items.

First, depending on the number of items that the teacher needs to generate, the diagram initially has randomized digits generated, integrating the Fisher-Yates Algorithm into the application.

The application compares every item in the array with O(n) complexity [9] to verify the legitimacy of the generated items.

If a unique pair of items is found, it will remain saved in the database; if a duplicate pair of items is found in the array, it will remove the items and replace them with a new unique pair of digits.

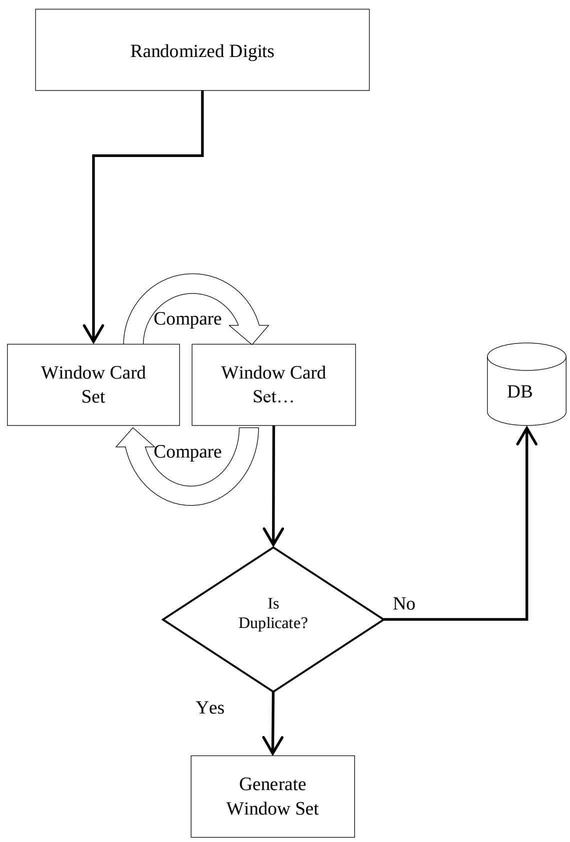

Fig. 3 Diagram Eliminating and Replacing Duplicate Randomized Digit Window Set

4.3. Plotting the Generated Window Card Set to Learners

Figure 4 shows the illustration of how the teacher performs an assessment for learners using the generated window card. The teacher selects a quarter and chooses the generated window card to be assisted at this point.

A teacher can configure how long the assessment takes. After setting the date and time limit to answer the questions from the learners, the test can be started on the specified date and time, once the learners access the active window card assessment.

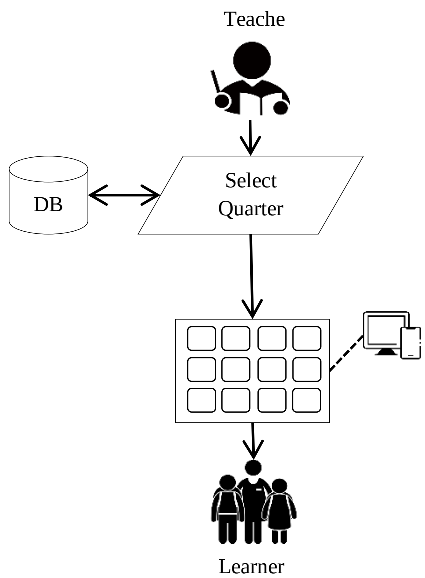

Fig. 4 Illustration of the Plotting of the Generated Window Card to learners

4.4. Schema of the Window Card Generator Application

Figure 5 shows the database schema of the Window Card Generator Database. It was understood that the design of the database is fully developed, considering the implementation of database normalization and properly assigned primary keys and foreign keys on each table. The schema is composed of 17 tables and well-assigned relationships.

The relation covers the registration and enrolment of learners, including schedules and subjects enrolled by the learners. On the other hand, the design of the database can also handle the recording of scores of each learner who took the test. It can also evaluate and produce statistical analysis and item analysis, where the items can be categorized into the least learned and most learned.

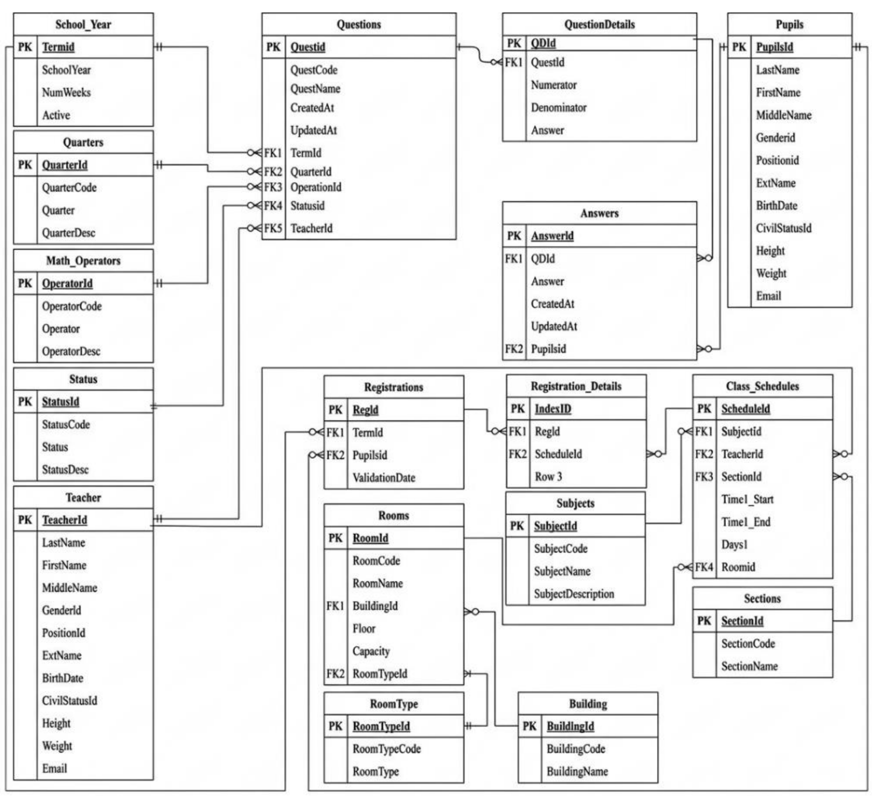

Fig. 5 Entity Relationship Diagram of the Generating Window Card for Elementary School Learners

4.5. Statistical Treatment

The Descriptive Statistical formula is used to get the population of the learners of Grade 6 in Brgy. Doclong 2nd, San Clemente, Tarlac. The formula to get the mean is:

$$\mu = \frac{\sum X}{N}$$

where:

$\sum X$ = The sum of all the values in the population
X = Learners

5. Results and Discussion

5.1. Setup

The hardware used for the experimental apparatus consists of 1TB of hard Disk, an AMD Ryzen 3 5300U CPU @ 2.60 GHz, Installed Memory of 16.0 GB, and it is running Windows 11 Operating System with the MySQL Server installed. Using the Fisher-Yates Algorithm to generate an unbiased permutation of paired numbers. The researcher also performed numerous generations of unbiased paired numbers.

5.2. Application Interface

5.2.1. LogIn Interface

Figure 6 shows the login interface of the Windows Card Generator. This module allows the different users of the system to log in before navigating their different functions in the system.

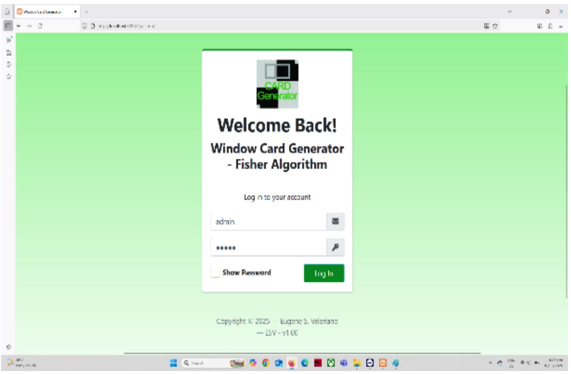

Fig. 6 Login Interface of Window Card Generator

5.2.2. Teacher Interface

Figure 7 shows the teacher interface of the Window Card Generator. This module allows the teacher to generate window cards of the different basic math operations, implementing the Fisher-Yates Shuffle algorithm.

A teacher is able to create cards in a specific quarter, number of items to be generated, grade level, sections, date of activity, time limit, and also the place values are configurable. Another significant feature of the teacher module is the ability to track learners who have already taken the activity and scores, as shown in Figure 8.

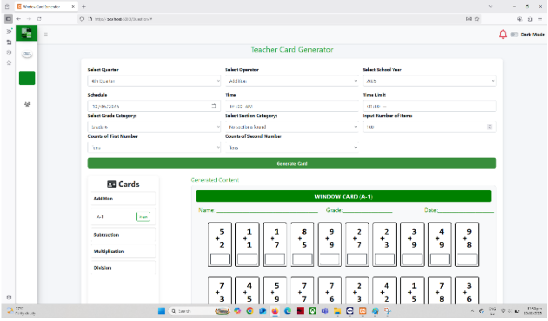

Fig. 7 Teacher Interface of Window Card Generator

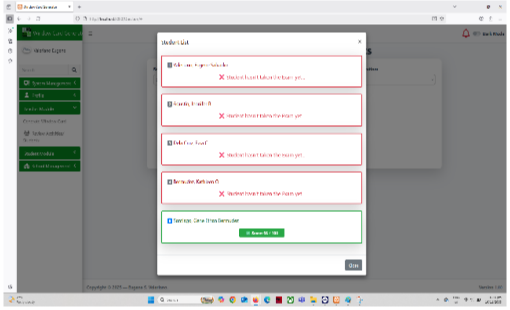

Fig. 8 Track Learners' Activities and Scores

5.2.3. Learner's Interface

Figure 9 shows the learner interface of the Window Card Generator. In this module, the learner sees the active activity given by the teacher, and the learner can start answering immediately once they click on the given activity. Additionally, the learner was able to review answers and check mistakes, and the system also provided the correct answer for it.

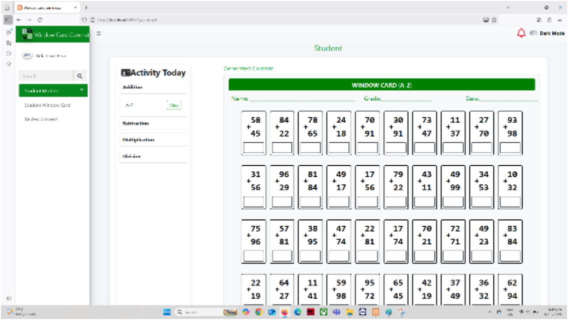

Fig. 9 Learner's Interface of Window Card Generator

5.2.4. Admin Interface

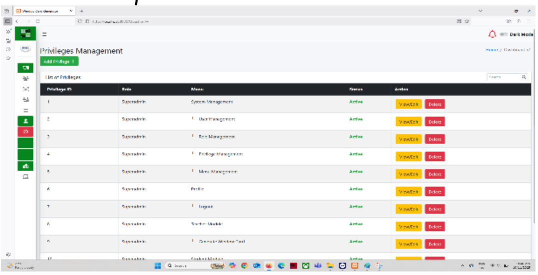

Fig. 10 Admin Interface of Window Card Generator

Figure 10 shows the admin interface of the Windows Card Generator. This module allows the admin to manage other essential components of the system, like managing teacher and student registrations, grade and section management, quarters, and user management.

5.3. Window Card

The Fisher-Yates Algorithm is used to generate the results of a permuted, unbiased collection of numbers that are integrated into the application.

5.3.1. Multiplication Operation

Table 1 shows the x and y variables, where the x variable stores all of the array's numerical contents that are iteratively divisible by 0, and the y variable stores the array's contents that are divisible by 1.

By using the technique in Figure 3 to evaluate every number produced by the algorithm, the permutation results demonstrate the unique paired numbers and ensure that there are no duplicate paired numbers.

Table 1. Permutation number produced by the algorithm in the multiplication operation

Variable	Permutation
x	35,81,84,67,3,10,74,19,91,33,21,80,13,53,95,32,27,73,94,96,62,45,70,0,30,16, 38,78,69,14,59,71,23,5,39,29,77,50,42,79,15,68,26,76,18,85,65,24,22,54,9,32,20,57, 86,49,50,61,43,18,98,66,79,60,11,84,97,67,30,16,78,42,83,37,63,81, 74,15,80,21,55,99,36,59,4,45,87,70,54,65,51,41,7,35,88,10,62,26,33,48
y	37,60,97,44,34,58,46,4,40,87,82,83,93,7,86,90,55,99,20,72,49,48,75,12, 47,41,88,6,52,63,89,43,36,25,11,28,66,8,57,31,98,9,17,2,51,1,92,56,64,61,77, 13,95,53,38,93,96,27,34,64,25,17,69,6,3,58,85,8,24,2,39,76,31,82,22,0, 14,19,29,94,23,56,46,52,1,75,89,47,68,91,92,72,28,90,5,40,12,71,73,44

5.3.2. Division Operation

Table 2, however, is a similar method applied to the multiplication operation, where variable x stores all of the array's numerical contents that are iteratively divisible by 0, and the y variable stores the array's contents that are divisible by 1. Using the methods from Figure 3, eliminate the duplicate paired numbers and make sure to produce unique paired numbers.

Table 2. Permutation count produced by the division operation utilizing the Algorithm

Variable	Permutation
x	35,81,84,67,3,10,74,19,91,33,21,80,13,53,95,32,27,73,94,96,62,45,70,0,30,16,38,78,69,14,59,71, 23,5,39,29,77,50,42,79,15,68,26,76,18,85,65,24,22,54,9,32,20,57,86,49,50,61,43,18,98,66,79,60, 11,84,97,67,30,16,78,42,83,37,63,81,74,15,80,21,55,99,36,59,4, 45,87,70,54,65,51,41,7,35,88,10,62,26,33,48
y	37,60,97,44,34,58,46,4,40,87,82,83,93,7,86,90,55,99,20,72,49,48,75,12,47,41,88,6,52,63,89, 43,36,25,11,28,66,8,57,31,98,9,17,2,51,1,92,56,64,61,77,13,95,53,38,93,96,27,34,64,25,17,69, 6,3,58,85,8,24,2,39,76,31,82,22,0,14,19,29,94,23,56,46,52,1,75, 89,47,68,91,92,72,28,90,5,40,12,71,73,44

5.3.3. Addition Operation

Table 3 shows that the elements of an array that are divisible by 0 throughout its contents are represented by the variable x, and the elements of an array that are divisible by 1 throughout its contents are represented by the variable y.

Each pair of numbers is uniquely recognized using the permutation results generated by the algorithm, which was evaluated and found to have no duplicate set of windows repeated using the technique shown in Figure 3.

Table 3. Permutation count produced by the addition operation utilizing the Algorithm

Variable	Permutation
x	35,81,84,67,3,10,74,19,91,33,21,80,13,53,95,32,27,73,94,96,62,45,70,0,30,16,38,78,69,14, 59,71,23,5,39,29,77,50,42,79,15,68,26,76,18,85,65,24,22,54,9,32,20,57,86, 49,50,61,43,18,98,66,79,60,11,84,97,67,30,16,78,42,83,37,63,81,74,15,80,21,55,99,36,59, 4,45,87,70,54,65,51,41,7,35,88,10,62,26,33,48
y	37,60,97,44,34,58,46,4,40,87,82,83,93,7,86,90,55,99,20,72,49,48,75,12,47,41,88,6,52, 63,89,43,36,25,11,28,66,8,57,31,98,9,17,2,51,1,92,56,64,61,77,13,95,53,38,93,96,27,34,64,25, 17,69,6,3,58,85,8,24,2,39,76,31,82,22,0,14,19,29,94,23,56,46,52,1, 75,89,47,68,91,92,72,28,90,5,40,12,71,73,44

5.3.4. Subtraction Operation

Table 4 displays the x and y variables. The x variable saves all of the array's numerical contents that are iteratively divisible by 0, and the y variable records the array's contents that are divisible by 1. By using the technique in Figure 3, the permutation results display the unique paired number. This ensures that there are no duplicate numbers among all the numbers produced by the process.

Table 4. Permutation number produced by the algorithm in the subtraction operation.

Variable	Permutation
x	35,81,84,67,3,10,74,19,91,33,21,80,13,53,95,32,27,73,94,96,62,45,70,0,30,16,38,78, 69,14,59,71,23,5,39,29,77,50,42,79,15,68,26,76,18,85,65,24,22,54,9,32,20,57,86, 49,50,61,43,18,98,66,79,60,11,84,97,67,30,16,78,42,83,37,63,81,74, 15,80,21,55,99,36,59,4,45,87,70,54,65,51,41,7,35,88,10,62,26,33,48
y	37,60,97,44,34,58,46,4,40,87,82,83,93,7,86,90,55,99,20,72,49,48,75,12,47,41,88,6, 52,63,89,43,36,25,11,28,66,8,57,31,98,9,17,2,51,1,92,56,64,61,77, 13,95,53,38,93,96,27,34,64,25,17,69,6,3,58,85,8,24,2,39, 76,31,82,22,0,14,19,29,94,23,56,46,52,1,75,89,47,68,91,92,72,28,90,5,40,12,71,73,44

5.4. Result Comparison from Traditional Window Card and a generated random window card using the Fisher-Yates Shuffle Algorithm.

Table 5 presents the total scores of learners across the four fundamental mathematical operations, comparing the performance outcomes using the traditional window card and the generated window card developed through the Fisher-Yates Algorithm.

The data were obtained from Grade 6 learners of Doclong 2nd Elementary School under the supervision of their class adviser. This assessment aimed to determine the differences and instructional impact of employing the generated window card. The results indicate a statistically significant difference between the two approaches, wherein learners generally obtained higher scores with the traditional window card compared to the generated version. These findings are further illustrated in Figure 11.

Table 5. Total Score Result of Learners Per Operation. (a)Traditional, and (b) Generated.

(a) Traditional Result

Learners	Operators			
	X	+	-	/
	Traditional	Traditional	Traditional	Traditional
1	100	100	100	90
2	100	100	100	72
3	100	100	100	46
4	99	99	99	15
5	99	99	100	38
6	99	100	98	40
7	99	99	100	74
8	96	100	100	32
9	95	100	87	52
10	95	97	93	35
11	92	98	100	31
12	92	99	98	18
13	88	100	96	28
14	64	98	99	30
15	50	99	93	22
16	38	99	56	18
17	21	86	22	11
18	9	100	43	7
Grand Mean	**79.78**	**98.50**	**88.00**	**36.61**

(b) Generated Result

Learners	Operators			
	X	+	-	/
	Generated	Generated	Generated	Generated
1	100	100	100	85
2	95	100	100	70
3	95	100	100	40
4	90	95	98	12
5	95	98	98	35
6	90	99	96	35

7	96	98	100	70
8	93	97	100	30
9	90	100	85	50
10	92	95	90	32
11	90	95	99	30
12	91	99	95	15
13	85	100	94	27
14	60	97	91	28
15	40	99	89	20
16	35	99	50	15
17	21	80	20	10
18	8	98	40	5
Grand Mean	**75.89**	**97.17**	**85.83**	**33.83**

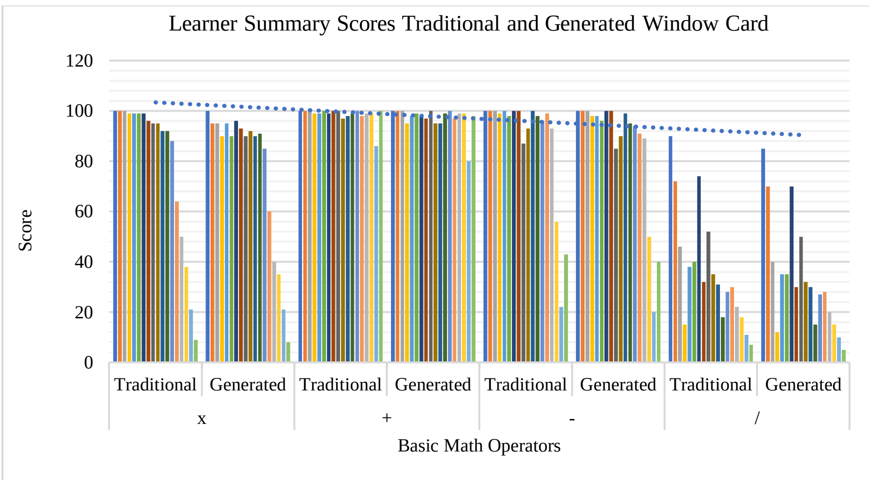

Fig. 11 Learners' Scores Graphical Presentation by Basic Math Operations

Figure 11 presents the trend in learners' performance across the four mathematical operations. The results indicate that learners encountered the greatest difficulty with the division operation, obtaining a grand mean average percentage of 36.61% to traditional method, while 33.83% for the generated method. This was followed by the multiplication operator, with a grand mean average percentage of 79.78% traditional method, while 75.89% for the generated. Subtraction operation ranked next in terms of difficulty, with a grand mean average percentage of 88% to the traditional method, while in the generated method, it is 85.83%. Conversely, the results reveal that addition was the least challenging for learners, as reflected by the highest grand mean average of all learners by 98.50% in the traditional method, while in the generated method, it is 97.17%.

6. Conclusion

In summary, the Fisher-Yates algorithm is very useful in implementing and incorporating the shuffling mechanism to generate randomized numbers for window card generation of the basic math operations for assessing learners in Elementary School. In addition, researchers used the algorithm to improve the efficiency and difficulty of the test items. Finally, the algorithm has a significant contribution to make the question more engaging to learners.

Recommendation

Wide implementation of the Fisher-Yates algorithm for shuffling to generate a window card for Elementary School learners to utilize the generated numbers and assess the mastery of the learners in the four basic operations. Report generation should be available as needed by the teacher, like a summary of scores, least learned and most learned items, item analysis, and numeracy level.

Acknowledgement

The researcher would like to thank everyone who helped make the study a success in one way or another. Thanks to Dr. Arnold R. Lorenzo, Dr. Leonell P. Lijauco, and Dr. Ruben A. Parazo for bringing the researchers' vision to life. To the mentors who helped the researcher develop the study by sharing their knowledge. To the teaching and non-teaching colleagues for their support, as well as to the researcher's family for their unwavering support. To our Almighty God for providing the researcher with the knowledge, wisdom, and skills necessary to complete the study.

References

[1] Medika Risnasari et al., "Computer based Test using the Fisher-Yates Shuffle and Smith Waterman Algorithm," *KnE Social Sciences*, vol. 2021, pp. 353-360, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Y. Haryanto, "Designing Educational Games with Fisher Yates Shuffle Method in Early Childhood at TKIT Al Muqorrobin based on Android," *Instal: Jurnal Komputer*, vol. 15, no. 2, pp. 414-423, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Imam Haditama, Cepy Slamet, and Deny Fauzy, "Implementation of the Fisher-Yates and Fuzzy Tsukamoto Algorithms in an Android-based Sundanese Tune Guessing Quiz Game," *Online Journal of Informatics*, vol. 1, no. 1, pp. 51-58, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[4] Chandra Kirana, Benny Wijaya, and Abdul Holil, "Implementation of the Fisher-Yates Shuffle Algorithm in Exam-Problem Randomization on M-Learning Applications," *Khazanah Informatika: Journal of Computer Science and Informatics*, vol. 7, no. 2, pp. 47-51, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Widi Aulia Rohmah, Asriyanik Asriyanik, and Winda Apriyandari, "Implementation of the Algorithm Fisher Yates Shuffle on Game Quiz Environment," *Journal of Informatics and Telecommunication Engineering*, vol. 4, no. 1, pp. 161-172, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Shafali Agarwal, "A Fractal based Image Cipher using Knuth Shuffle Method and Dynamic Diffusion," *International Journal of Computer Networks and Communications*, vol. 11, no. 4, pp. 81-100, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] Louie Salvador, Effectiveness Of Winc-I Intervention on Learners' Mastery of Operations on Integers of Grade 8 -Spa Students of Cavite National High School, 2024. [Online]. Available: https://www.researchgate.net/publication/380005928_EFFECTIVENESS_OF_WINC-I_INTERVENTION_ON_LEARNERS'_MASTERY_OF_OPERATIONS_ON_INTEGERS_OF_GRADE_8_-SPA_STUDENTS_OF_CAVITE_NATIONAL_HIGH_SCHOOL

[8] Kanika et al., "Effect of Different Grouping Arrangements on Students' Achievement and Experience in Collaborative Learning Environment," *Interactive Learning Environments*, vol. 31, no. 10, pp. 6366-6378, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Chung-Yang (Ric) Huang, Chao-Yue Lai, and Kwang-Ting (Tim) Cheng, *Fundamentals of Algorithms*, Electronic Design Automation, Elsevier Inc, pp. 173-234, 2009. [CrossRef] [Google Scholar] [Publisher Link]